Technical Report - **Product specification**

Volcanic Activity Monitoring System

Course:	IES - Introdução à Engenharia de Software
Date:	Aveiro, October 7, 2024
Students:	113655: Abel Teixeira 114321: José Marques 112959: Luís Godinho 113586: Tiago Lopes
Project abstract:	**Volcanic Activity Monitoring System (VAMS)** is based in collecting, analysing and presenting information of volcanic activity data obtained through an API. VAMS simplifies the user experience while enriching their engagement with up-to-date, actionable data.

Table of contents:

1 Introduction

The main objective of this project is to develop a fully functional, all-service application. In addition to the application itself, this project offers a comprehensive learning experience, covering key elements of application development such as user story creation, branch management, and agile architecture. This report outlines our journey toward accomplishing these goals.

2 Product concept

Vision statement

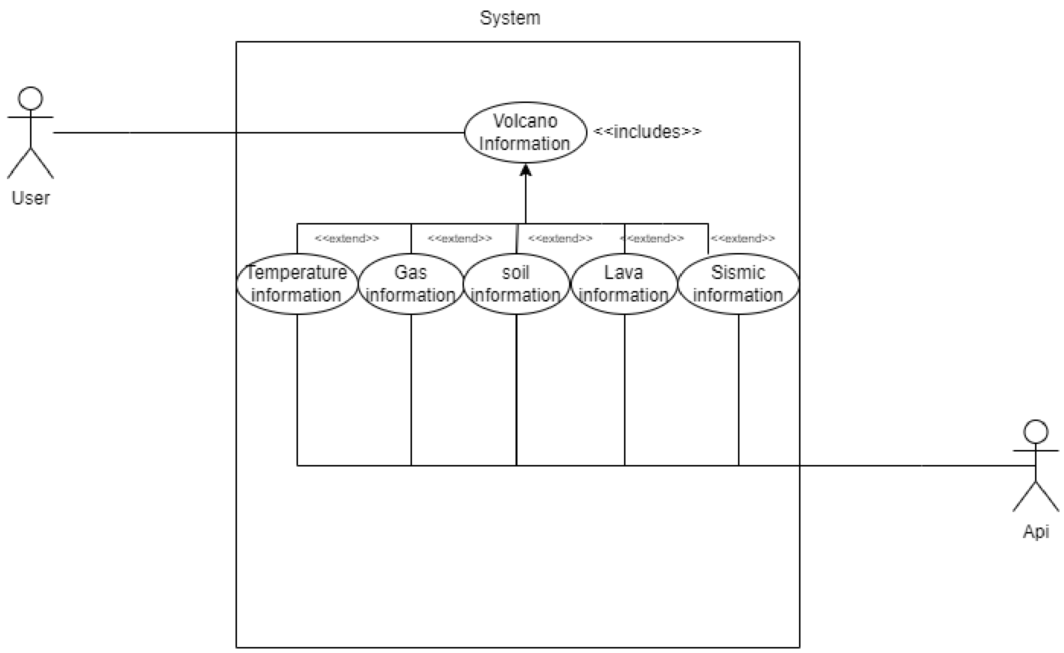
VAMS focuses on the collection, analysis, and presentation of volcanic activity data sourced through a reliable API. It addresses the critical need for a more accessible and informative way of monitoring and understanding volcanic phenomena. By providing a seamless solution for gathering, organising, and displaying real-time volcanic data, VAMS enables users, whether scientists, enthusiasts, or safety authorities to stay informed about potential volcanic hazards, geological changes, and eruption predictions.

In the field of volcanology, users often struggle with accessing up-to-date data. VAMS solves this by delivering real-time notifications, detailed visualisations, and user-friendly tools that simplify the interpretation of complex volcanic activity.

While several monitoring systems exist, VAMS distinguishes itself through its intuitive interface and comprehensive approach, promoting both safety and education. It is designed to be more than just a data provider; it is an interactive platform that empowers users to engage with volcanology more effectively, making informed decisions in response to volcanic risks.

One of the most distinguishing features implemented in our product is the ability to add volcanoes to monitor. For that purpose, all the user needs to do is to name the volcano, set its coordinates, add a description (optionally) and import their sensors. After that, the volcano is automatically added to the database and all its data will start to show up in the diagrams.

Use-Case Diagram



Personas and Scenarios

Personas

Carlos Mendoza is a 38 years old Emergency Response Coordinator, he has a basic proficiency in web and mobile apps and his main goals are to ensure the safety of communities living near volcanoes and to communicate risks and evacuation plans effectively to local authorities and citizens.

For this he needs a simple and understandable visualisations of volcanic activity and a way to communicate complex data to non-technical audiences

Dr. Emily Reyes is a 45 years old Senior volcanologist at a research institute, she is proficient with data analysis software, GIS tools, and scientific programming languages. Her main goals are to study long-term patterns of volcanic activity and To access and analyse comprehensive data about volcanic gases, seismic activity, and other indicators of eruptions.

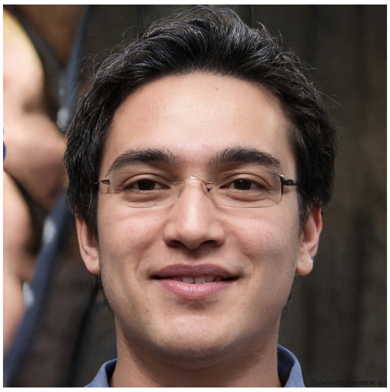
For this she needs real-time, detailed, and historical data to make accurate predictions and requires access to raw data to perform custom analyses and modelling.

Maria Gomez is a 30 years old Local Resident, she has an average smartphone proficiency and her main goals are to stay informed and receive alerts about near volcanic activity.

For this she needs timely information on whether the volcano poses any risk.

Sarah Patel is a 32 year old Environmental Scientist who studies the environmental impact of volcanic activity on local ecosystems, she is familiar with data analysis tools and environmental monitoring software. Her main goals are to track changes in gases and ground composition and vegetation health over time and to collaborate with other researchers on environmental impact assessments.

For this she needs detailed gas composition data to analyse environmental impact



Jason Lee is a 27 year old Data Analyst, he has advanced data science skills and proficiency in programming languages.

His main goals are to access data on seismic activity and gas emissions to develop a predictive model for volcanic eruptions.

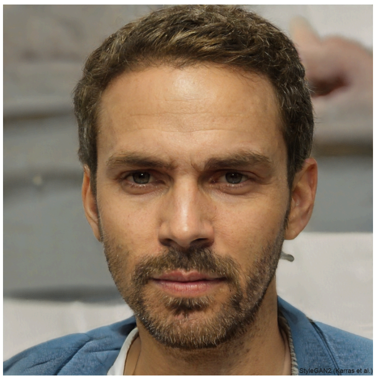
For this he needs access to real-time and historical data and requires tools for data visualisation.

Aisha Thompson is a 22 year old, Geology Student who has an interest in volcanology, she is familiar with basic research tools, good at using web applications. Her main goals are to learn about volcano monitoring methods and data interpretation and to complete research projects for her degree using real-world data.

For this she needs access to simplified explanations of volcanic data and terms.

David Kim is a 40 years old Journalist who is familiar with web applications and is able to work with basic visual and multimedia tools. His main goals are to get accurate information on volcanic activity for reporting purposes.

For this he needs real time information to report breaking news.



Anacleto José is a 34 year old, he is a Tour Guide, who tours on an island where there is active volcanic activity, he is comfortable with using web apps. His main goals are to ensure the safety of tourists while providing an engaging, educational experience and to share accurate and interesting information about the volcano's activity, history, and geological features with tourists.

For this he needs up-to-date volcanic activity information in real time to avoid putting tourists in danger and to provide tourists with engaging stories about the volcano's activity and geological significance

Scenarios

Carlos, the lead emergency planner for a government agency, receives an alert during his morning briefing about rising seismic activity near **Santa Barbara volcano**, a volcano located near several populated villages. He enters on the volcano monitoring website on his tablet and sees that both seismic and gas emission levels have spiked above normal levels. He quickly checks the real-time risk assessment on the dashboard, which shows an increased likelihood of an eruption within the next 48 hours. Carlos the assess the situation and based on the information provided, Carlos coordinates an emergency response meeting with local officials

Dr. Emily Reyes is researching the **Santa Barbara volcano** . She's preparing a paper on how gas emissions, specifically sulphur dioxide (SO_2), correlate with eruption events. Using the volcano monitoring website, she pulls up historical data on gas emissions and seismic activity from the past 10 years for the volcano. To investigate further, Emily downloads the raw data and imports it into her data analysis software.

Maria lives with her family near **Santa Barbara volcano**. While cooking dinner, Maria receives a notification on her phone. The alert indicates that seismic activity around the volcano has increased to a yellow alert level, meaning potential danger but no immediate risk. Maria checks VAMS website, which uses a colour-coded system to show the current risk level: yellow. She then prepares for evacuation.

Sarah is an environmental scientist studying the impact of volcanic emissions on air quality and local ecosystems around **Santa Barbara volcano**. Recently, local vegetation near the volcano has been dying off, and Sarah suspects that changes in sulphur dioxide levels might be affecting plant health. She accesses the volcano monitoring app to retrieve data on SO_2 emissions over the past several months. Sarah overlays the SO_2 data with weather patterns and vegetation health reports.

Jason, a data analyst working with a volcano observatory, is tasked with creating predictive models to forecast eruptions of **Santa Barbara volcano**. He needs access to both real-time and historical data on seismic activity, ground deformation, and magma viscosity. Using the app, Jason downloads large datasets covering the past 30 years of volcanic activity. He imports the data into his machine learning software and begins building models to identify patterns that precede eruptions.

Aisha, an undergraduate geology student, is working on a project about the volcanic activity of **Santa Barbara volcano**. She needs to understand how gas emissions and seismic activity relate to eruptions. She enters the volcano monitoring website to access historical data from the volcano. The app provides a student-friendly interface with simplified explanations and visualisations. Aisha then downloads data on past eruptions, focusing on the seismic and gas data leading up to each event.

David is a journalist writing a story about the recent activity at **Santa Barbara volcano**. He needs real-time, reliable information to report on whether the volcano is close to erupting. David opens the volcano monitoring website and sees that seismic activity and gas emissions have both been increasing steadily over the past week. The app provides visual graphs with the information over time, which he downloads to include in his article.

Anacleto, a tour guide, is leading a group of tourists on a tour around **Santa Barbara volcano**. He frequently checks the volcano monitoring website to ensure the group's safety. During the tour, Anacleto receives an alert on his phone indicating a small increase in seismic activity. He pauses the tour and checks the website's real-time risk assessment, which shows that the risk is still low, but he decides to reroute the group to avoid areas near the crater.

Product requirements (User stories)

1. As an emergency response coordinator, I want to see a dashboard of key indicators (seismic activity, gas levels, ground movement) with easy-to-understand visuals so I can assess the current risk quickly.
2. As a volcanologist, I want to view real-time data on seismic activity and gas emissions for a selected volcano so that I can monitor potential signs of an eruption.
3. As a local resident, I want to see how close the activity or potential danger is to my home, so I can make decisions on whether to evacuate.
4. As an environmental scientist, I want to analyse gas emission trends to assess their impact on local air quality and ecosystems.
5. As an environmental scientist, I want to view ground composition data and changes over time, so I can study how volcanic activity affects soil health and plant life.
6. As a data analyst, I want to visualise trends in ground deformation and magma viscosity over time, so I can identify key indicators of future eruptions.
7. As a geology student, I want to access simplified explanations of volcanic data (e.g., magma viscosity, seismic activity) so that I can better understand how volcanic monitoring works.
8. As a journalist, I want to embed maps, graphs, or data visualisations from the platform into my articles or reports, so I can present information in a visually appealing way.
9. As a tour guide, I want to share interesting facts about the volcano's recent activity, eruption history, and geological details with my group, so I can provide an educational and exciting experience.

3 Architecture notebook

Key requirements and constraints

As a real time data visualisation system, our main architectural focus is to build a system which can provide a flow from data generation to user presentation with as little latency as possible.

- The system must have a web interface compatible with most browsers.
- The webpage's interface must be simple to use.
- The components must be able to scale independently from each other.
- The site must be able to respond properly even when dealing with high traffic.
- The system must be able to update the database with said data.
- The webpage must display nearby seismic activity.
- The system must not allow unauthorised access to the database.

Architectural view

Front-End (Web App):

- **User Interface (UI)**: The web application is the main point of interaction for users. It communicates with the back-end.
- **Communication**: Data is exchanged between the front-end and back-end is possible using a bidirectional HTTP connection.

Back-End:

- **REST API Controller:** This component is responsible for handling requests from the front-end and processing of the appropriate responses. It is the main interface for communication between the client side application and the system's back-end logic.
- **Data Management Service**: Manages the flow of data between the API controller and the database. It may also handle complex data processing tasks.
- **Communication Flow**: The backend should not only retrieve data from the database to display on frontend but also save the data incoming from the generators.

Database(ScyllaDB):

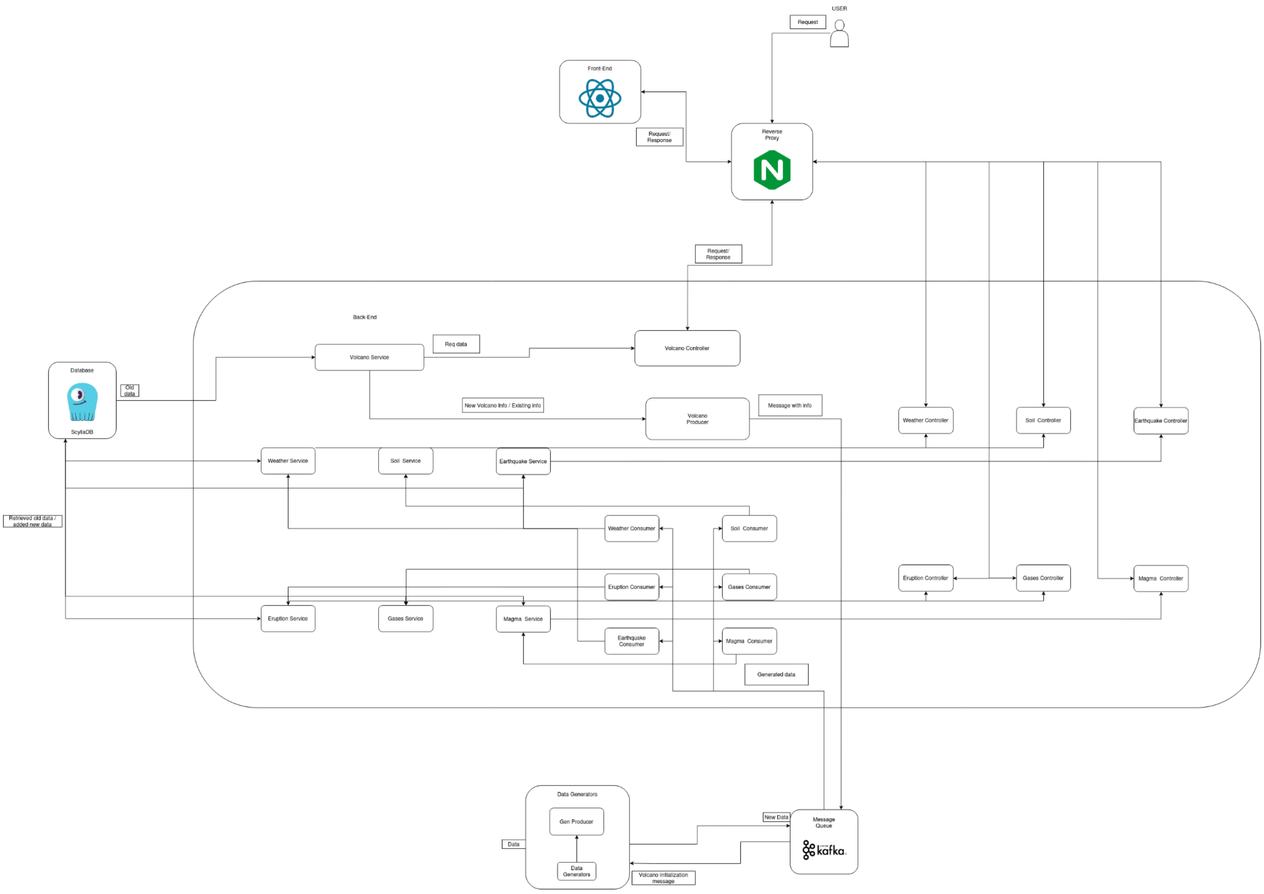
- **High Performance**: ScyllaDB is optimized for speed and can handle millions of operations per second, making it suitable for time-series workloads.
- **Time-Series Optimization**: ScyllaDB's architecture excels in handling large volumes of sequential writes and reads, which are typical in time-based datasets like logs, metrics, and event tracking.

Data Sources:

- **Data Generation Service: T**his component produces real-time information and transmits it to the back-end through a message queue for processing.
- **Message Queue:** The message queue serves to split information producers (the data generation service) from information consumers (the back-end), allowing asynchronous communication among them.

Flow and Interaction:

- When starting the application, the backend send to the generators, all existing volcanoes, so their generators can be initialized
- The data generation service communicates with the message queue, which in turn sends data to the back-end.
- The back-end processes the incoming data, stores the necessary parts in the database, and sends relevant data to the front-end, through websockets if open.


USER
Request
Front-End
Request/
Response
Reverse
Proxy
Request/
Response
Back-End
Database
ScyllaDB
Old
data
Volcano Service
Req data
Volcano Controller
New Volcano Info / Existing info
Volcano
Producer
Message with info
Weather Controller
Soil Controller
Earthquake Controller
Weather Service
Soil Service
Earthquake Service
Retrieved old data /
added new data
Weather Consumer
Soil Consumer
Eruption Controller
Gases Controller
Magma Controller
Eruption Consumer
Gases Consumer
Eruption Service
Gases Service
Magma Service
Earthquake
Consumer
Magma Consumer
Generated data
Data Generators
Gen Producer
Data
Data
Generators
Volcano initialization
message
New Data
Message
Queue
kafka

Module interactions

1. Data Sources

- **Function**: These represent external systems or devices that generate new real-time data.
- **Flow**: New real-time data is sent to the **Message Queue** component.

2. Message Queue

- **Function:** The Message Queue acts as a buffer that gets real-time information from the data sources and guarantees asynchronous communication among the data sources and the back-end system. It allows decoupling between the data generation and data processing.
- **Flow**:
    - Generators initialization:
        - **Backend** sends all volcanoes or the one created
        - **Consumer** receives the relevant information
        - Initializes the generators with that information
    - Data creation:
        - Receives data from **Data Sources**.
        - Sends a **Request for data** to the **Back End**.

3. Back End

- **Function**: This is the central processing component that handles requests for data, processes it, and interacts with the database. It performs operations like inserting new data or requesting old data from the database.
- **Flow**:
    - Receives the **Request data** from the **Message Queue**.
    - Sends **Request data** to the **Database** to fetch either old or new data as needed.
    - Receives **Old data** from the **Database**.
    - Sends **Latest data retrieval** to the **Front End** for display.
    - Inserts new data into the **Database**.

4. Database

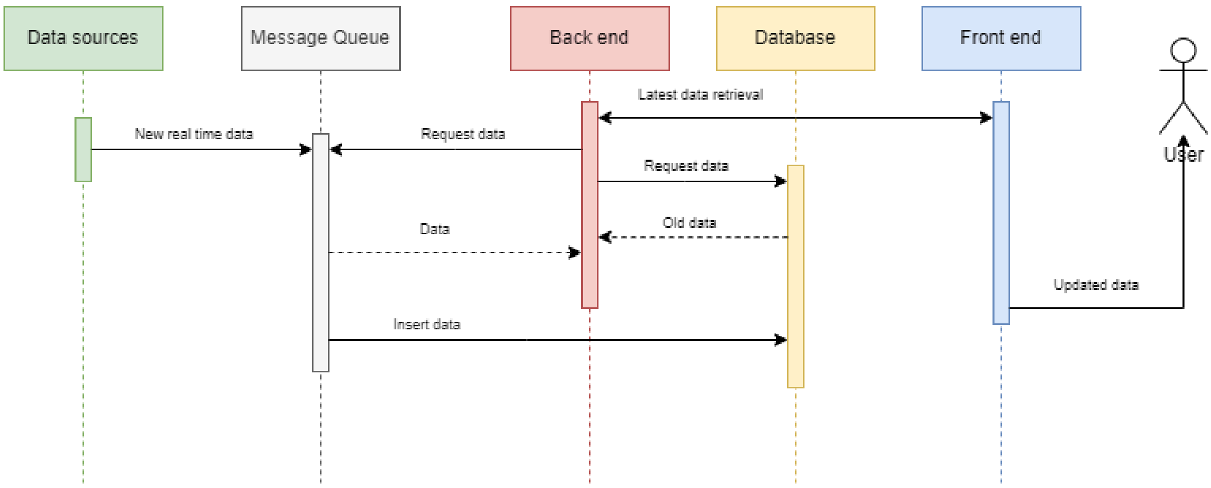
- **Function**: The database stores and manages both old and new data. It responds to requests for data retrieval and allows data to be inserted.
- **Flow**:
    - Receives **Request data** from the **Back End**.
    - Sends **Old data** (stored data) back to the **Back End** for processing.

5. Front End

- **Function**: The front end is the interface that interacts with the user. It displays the updated data retrieved from the back end, allowing the user to see the latest information.
- **Flow**: Receives **Latest data retrieval** from the back end and presents **Updated data** to the user.

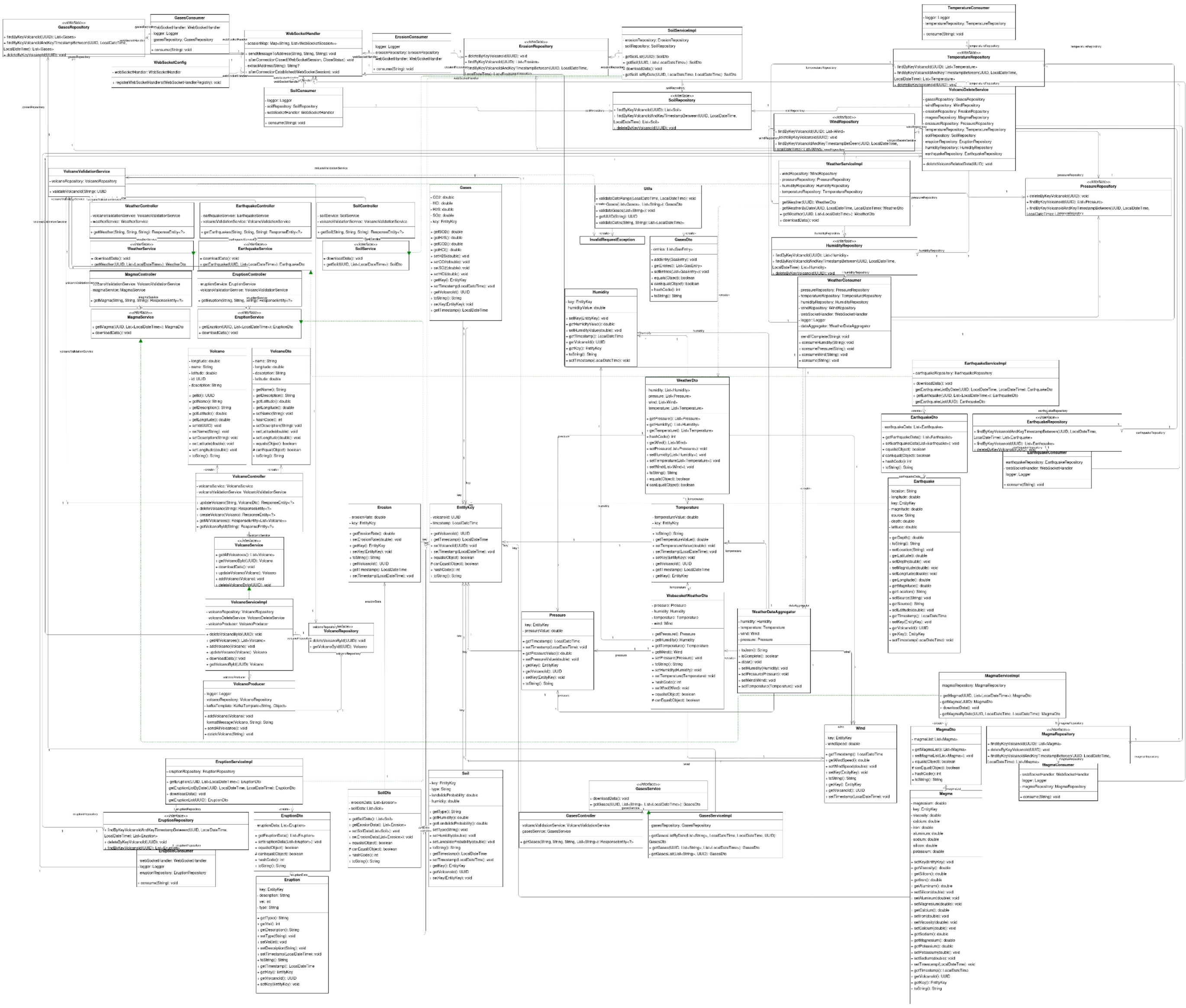
6. User

- **Function**: The end-user interacts with the front end, viewing the updated data as processed by the system.



4 Information perspective

Class Diagram



Entities

Magma

Represents the composition and properties of magma, including:

- **Silicon, Iron, Aluminium, Calcium, Sodium, Magnesium, Potassium:** The chemical elements present in magma, each represented as a `double` value.
- **Viscosity:** The viscosity of magma, determining its flow resistance.
- **Timestamp:** The date and time when this magma data was recorded.

Gases

Represents gases emitted from volcanic activities, with data for:

- **H2S, CO2, SO2, HCl:** The concentrations of hydrogen sulphide, carbon dioxide, sulphur dioxide, and hydrochloric acid gases, respectively, recorded as `double`.
- **Timestamp:** Date and time when the gas data was recorded.

Pressure

Represents the atmospheric pressure:

- **pressureValue:** The value of pressure measured.
- **Timestamp:** Date and time when the pressure value was recorded.

Humidity

Represents the humidity levels:

- **humidityValue:** Humidity percentage in the atmosphere.
- **Timestamp:** Date and time of humidity data capture.

Temperature

Represents the atmospheric temperature:

- **temperatureValue:** The temperature measured in degrees.
- **Timestamp:** Date and time when the temperature data was recorded.

Wind

Represents wind speed:

- **windSpeed:** Speed of wind at a specific location.
- **Timestamp:** Date and time when the wind speed was measured.

Eruption

Represents volcanic eruption data:

- **year:** The year the eruption occurred.
- **type:** Type of eruption (e.g., explosive, effusive).
- **VEI:** Volcanic Explosivity Index, a number describing the magnitude of the eruption.
- **Description:** A textual description of the eruption.

Earthquake

Represents seismic activity information:

- **Date:** The date of the earthquake.
- **Latitude, Longitude:** Coordinates of the earthquake's location.
- **Depth:** The depth at which the earthquake occurred.
- **Magnitude:** The magnitude of the earthquake on the Richter scale.
- **Location:** A general description of the location.
- **Source:** The source of the earthquake data.

Soil

Represents soil data relevant to volcanic activity:

- **Type:** Soil type, which can impact volcanic effects.
- **Humidity:** Soil humidity percentage.
- **landslideProbability:** The likelihood of a landslide occurring.
- **Timestamp:** The date and time when the soil data was recorded.

Erosion

Represents the rate of soil erosion:

- **ErosionRate:** Rate at which the soil is being eroded.
- **Timestamp:** The date and time when erosion data was captured.

Services

VolcanoService

- **getVolcanoes():** Get all the volcanoes in the database.
- **getVolcano(volcanoId):** Get volcano by its id
- **AddVolcano(volcano):** Receives a new volcano, adds it to the database and sends it to the message queue so the generators can be initialized
- **EditVolcano(newVolcano):** If the volcano exists it edits its entries
- **DeleteVolcano(volcanoId):** Deletes the volcano and its entries from the database and stops the generators thread execution related to the volcano

VolcanoDeleteService

Responsible for deleting the volcano and its related data in an atomic way.

- **deleteVolcanoRelatedData**(volcanoId): deletes the data related to the volcanoId, before deleting the volcano.

EarthquakeService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, otherwise it retrieves all data.

EruptionService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, else it retrieves all data.

GasesService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, else it retrieves all data.

MagmaService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, else it retrieves all data.

SoilService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, else it retrieves all data.

WeatherService

- **getData(volcanoId, dateRange):** retrieve the data related to the volcanoId within the given dateRange, if given, else it retrieves all data.

Consumers

These are responsible to consume the data generated by the generators and sent through the message queue, each data type (e.g Earthquake, Soil) has its own topic.

5 References and resources

https://react.dev

https://spring.io/projects/spring-boot